UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC H3C 3J7

UNE PROCÉDURE DE RÉSOLUTION DU PROBLÈME D'AFFECTATION MULTIMODE-MULTIPRODUIT PAR DÉCOMPOSITION SUR UN ESPACE PRODUIT

par

Jacques Guélat

Cette étude a été publiée grâce à une subvention du Fonds F.C.A.R. pour l'aide et le soutien à la recherche.

Université de Montréal

Centre de recherche sur les transports - Publication #488

Avril 1986

Dépôt légal - 3e trimestre 1986

RESUME

Le problème de transport à moindre coût de plusieurs produits sur un réseau multimodal est décrit. Une procédure de résolution est proposée. On démontre sa convergence globale et quelques résultats sont analysés.

Mots-clés : modèle normatif, affectation multimode-multiproduit, algorithmes de Frank-Wolfe et Gauss-Seidel.

INTRODUCTION

On considère ici un problème de transport de marchandises sur un réseau multimodal. La demande entre chaque point du réseau étant connue, il s'agit de transporter ces marchandises à moindre frais. Il est entendu qu'une autorité centrale contrôle la situation et désire minimiser ses propres frais (et non ceux des transporteurs). Ce genre de problème s'inscrit dans un contexte de planification stratégique, par exemple au niveau de la politique de transports de marchandises d'un pays ou d'un organisme contrôlant plusieurs modes de transports. C'est donc un modèle *normatif* qu'on utilisera dans cette situation.

La planification étant gérée par un office central, une certaine répartition modale de la demande peut être connue ou imposée à priori. Le modèle doit donc être assez souple pour permettre le choix de sous-réseaux pour le transport des marchandises, ces sous-réseaux pouvant différer d'un produit à l'autre. Le choix de la représentation du réseau joue ainsi un rôle central et fait l'objet du premier chapitre de cet exposé.

Dans le second chapitre, nous énonçons formellement le problème d'affectation multimode-multiproduit. Le chapitre suivant est consacré à la description de l'algorithme proposé pour la résolution du problème. Une preuve de convergence est fournie au chapitre 4. Finalement, quelques résultats préliminaires obtenus lors du développement du logiciel sont exposés dans le dernier chapitre.

REPRESENTATION DU RESEAU

Pour fixer les idées qui vont suivre, introduisons le réseau trimodal suivant:

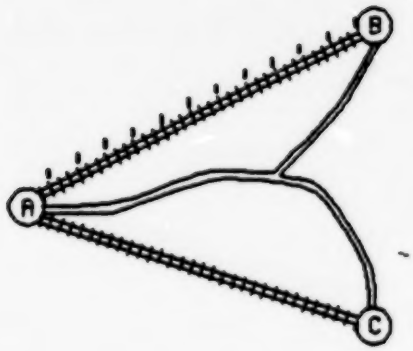

Figure 1. Réseau physique.

On y distingue trois modes de transport : la route et le chemin de fer diesel et électrique. Les villes A et B sont reliées par les trois modes, A et C par la route et le chemin de fer diesel, tandis que B et C ne peuvent communiquer que par la route.

La représentation la plus compacte d'un réseau multimodal consiste à ne considérer le mode que comme un attribut du lien. Le réseau de la figure 1 est alors representé ainsi :

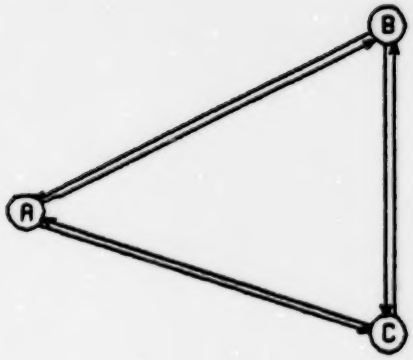

Figure 2. Représentation compacte.

L'arc (C,B) possède le mode routier comme attribut; l'arc (A,B) quant à lui possède les trois modes de transport.

Cette représentation nécessite la définition d'une fonction de coût sur le lien agrégée qui ne peut représenter la spécificité de chaque mode du lien, le flot de marchandises de chaque mode n'étant pas connu. D'autre part, l'indépendance physique des liens n'apparaît pas dans une telle représentation, ce qui est très gênant au niveau de la modélisation de la congestion d'un mode, par exemple.

Pour pallier à ces difficultés, il est nécessaire de connaître la distribution du flot de marchandises pour chaque mode. Ceci revient à conserver autant de copies du réseau qu'il n'y a de modes (si l'on veut rester efficace dans la modification des modes reliant deux noeuds, par exemple). Cette structure devient vite prohibitive avec la taille du réseau.

La représentation fidèle du réseau physique est un pas vers une solution praticable de ces problèmes. Dans une telle représentation, le réseau de le figure 1 prendrait la forme:

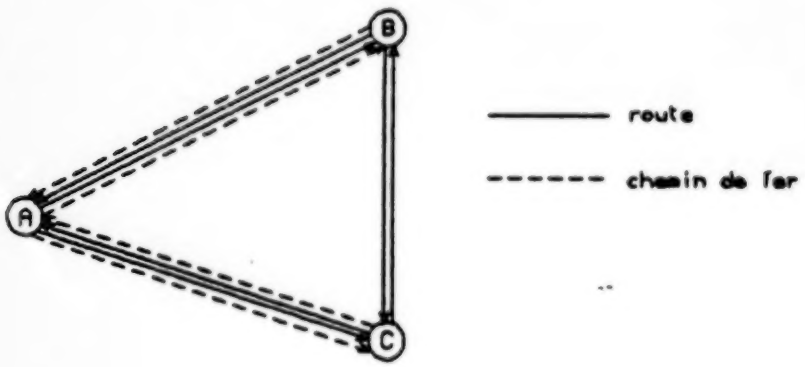

Figure 3. Représentation physique.

Cette représentation ne règle toujours pas la cas du lien ferroviaire (A,B) où peuvent circuler aussi bien des trains électriques que diesels. Nous en sommes donc amenés à associer à chaque lien du réseau un seul mode, ce qui est la façon la plus naturelle de représenter un réseau multimodal (voir figure 4). Cette dernière représentation permet d'associer avec chaque lien une fonction de coût bien spécifique puisqu'elle permet un accès direct au volume par mode. Elle n'est par contre pas économique quant au nombre total de liens.

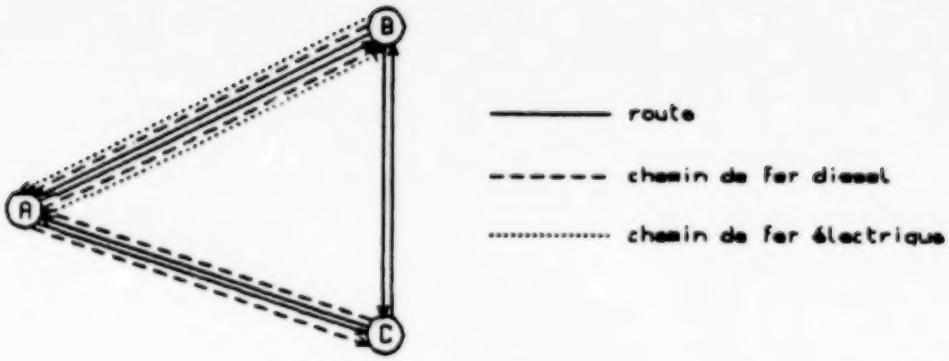

Figure 4. Représentation adoptée.

Dans cette représentation, le mode fait partie inhérente de la structure du réseau et n'est plus qu'un simple attribut. Ainsi, le parallélisme apparent introduit en passant de la figure 2 à la figure 4, qui est un phénomène indésirable dans nombre d'algorithmes de réseaux, n'en est en fait pas un puisque deux liens portant deux modes distincts sont structurellement différents (un lien est défini comme un triplet (origine,destination,mode)).

Quelle que soit la représentation choisie, il est nécessaire, si la répartition modale n'est pas connue, de pouvoir autoriser ou pénaliser certains transferts mode à mode aux noeuds du réseau. Ceci peut se réaliser en faisant exploser un sommet où des transferts ont lieu en autant de sommets qu'il n'y a d'arcs entrants et d'arcs sortants et en introduisant le graphe bipartit qui les relie:

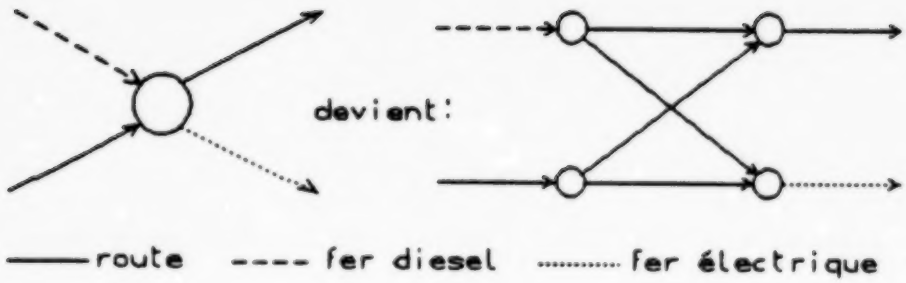

Figure 5. Explosion d'un noeud.

Cette méthode de faire possède les deux inconvénients suivants:

- elle fait augmenter considérablement le nombre d'arcs et de sommets;
- les nouveaux sommets créés lors de l'explosion n'ont pas de situation géographique propre.

De plus, dans le contexte de la représentation choisie auparavant, les nouveaux arcs créés ne possèdent pas à proprement parler un seul mode, mais bien deux.

Nous caractériserons un transfert par la donnée d'une paire d'arcs, l'un entrant dans le sommet où à lieu le changement de mode et un arc sortant de ce sommet. Ainsi on n'ajoute aucun nouveau lien ou sommet au réseau existant.

FORMULATION DU PROBLEME

Le principe normatif, qui s'applique dans le contexte du transport régi par une autorité centrale, consiste à minimiser le coût généralisé total occasionné à cette autorité par le transport de marchandises dans le réseau. La notion de coût est donc centrale dans la formulation de ce problème. Dans ce qui suit, nous allons tenter de lui conserver sa forme la plus générale.

Notons N l'ensemble des noeuds, A l'ensemble des arcs, T l'ensemble des transferts ($T \subseteq AxA$) et M l'ensemble des modes du réseau. Les cardinalités respectives de ces ensembles sont notées n_N, n_A, n_T et n_M. A chaque arc $a \in A$, on associe une *fonction de coût* s_a qui dépend du volume de marchandises traversant cet arc, ou plus généralement du volume global des marchandises transitant dans le réseau. On associe de même une fonction s_t avec chaque transfert $t \in T$.

Pour simplifier le problème, chaque marchandise est classée selon un type standard. L'ensemble de ces types s'appelle ensemble de *produits* et est dénoté P (de cardinalité n_P). Chacun de ces produits doit être transporté entre les diverses origines ($O \subseteq N$) et destinations ($D \subseteq N$) du réseau, ceci en utilisant certains modes. La demande entre chaque paire origine-destination pour chaque produit est fournie par un ensemble de matrices reflétant la répartition modale donnée à priori. Notons $g^{m(p)}$ une matrice de demande associée au produit $p \in P$, où $m(p)$ appartient à $M(p)$ l'ensemble de sous-ensembles de modes utilisés pour transporter le produit p.

Un flot v^p de produit p à travers le réseau est un vecteur de $E^{n_A+n_T}$ satisfaisant aux contraintes habituelles de conservation de flot et de non-négativité:

$$v^p = \begin{pmatrix} (v^p_a)_{a\in A} \\ (v^p_t)_{t\in T} \end{pmatrix} \in E^{n_A+n_T}$$

Le flot de marchandises dans le réseau est alors décrit par le vecteur $v = (v^p)_{p\in P} \in E^{n_P(n_A+n_T)}$.

A un flot donné v correspondent des coûts unitaires par produit sur les arcs et les transferts, notés $s^p_a(v)$ et $s^p_t(v)$ respectivement. De même manière que pour les flots, nous noterons s^p les coûts unitaires pour le produit $p\in P$,

$$s^p = \begin{pmatrix} (s^p_a)_{a\in A} \\ (s^p_t)_{t\in T} \end{pmatrix} \in E^{n_A+n_T}$$

et $s = (s^p)_{p\in P} \in E^{n_P(n_A+n_T)}$ la fonction vectorielle de coût unitaire:

$$s : E^{n_P(n_A+n_T)} \longrightarrow E^{n_P(n_A+n_T)}$$

$$v \longmapsto s(v)$$

Le coût total sur l'arc $a\in A$ pour le produit $p\in P$ s'écrit alors $s^p_a(v)v^p_a$. De même, le coût total sur le transfert $t\in T$ pour le produit $p\in P$ s'écrit $s^p_t(v)v^p_t$. Le coût total à minimiser s'écrit alors :

$$\sum_{p\in P} \left[\sum_{a\in A} s^p_a(v)v^p_a + \sum_{t\in T} s^p_t(v)v^p_t \right]$$

ou vectoriellement: $s^T(v)v$.

Notons F la fonction à minimiser:

$$F : \Omega \subseteq E^{n_p(n_A+n_T)} \longrightarrow E \qquad (1)$$

$$v \longmapsto F(v) = s^T(v)v$$

où Ω est l'ensemble des flots réalisables satisfaisant à la demande. Nous allons caractériser plus précisément cet ensemble. Soit $K_{od}^{m(p)}$ l'ensemble des chemins reliant l'origine $o \in O$ à la destination $d \in D$ et utilisant les modes de $m(p) \in M(p)$, $p \in P$. Pour satisfaire à la demande, on doit alors respecter les contraintes suivantes:

$$\sum_{k \in K_{od}^{m(p)}} h_k = g_{od}^{m(p)} \quad , o \in O,\ d \in D,\ m(p) \in M(p),\ p \in P \qquad (2)$$

où h_k est le flot sur le chemin k. La non-négativité des flots s'exprime comme:

$$h_k \geq 0 \quad , k \in K_{od}^{m(p)},\ o \in O,\ d \in D,\ m(p) \in M(p),\ p \in P \qquad (3)$$

La conservation de flot étant implicite à la notion de flot sur les chemins, il suffit, pour terminer la caractérisation du domaine réalisable Ω, de fournir les relations entre les flots sur les arcs et les flots sur les chemins, à savoir:

$$v_a^p = \sum_{k \in K^p} \delta_{ak} h_k \quad , a \in A,\ p \in P \qquad (4)$$

avec: $K^p = \bigcup_{m(p) \in M(p)} \bigcup_{o \in O} \bigcup_{d \in D} K_{od}^{m(p)}$

$$\delta_{ak} = \begin{cases} 1 \text{ si } a \in k \\ 0 \text{ sinon} \end{cases}$$

$$v_t^p = \sum_{k \in K^p} \delta_{tk} h_k \quad , \ t \in T, \ p \in P \qquad (5)$$

avec: $\delta_{tk} = \begin{cases} 1 \text{ si } t \in k \\ 0 \text{ sinon} \end{cases}$

On entend que le transfert t appartient au chemin k si les deux arcs définissant t en font partie.

Le problème qu'on doit résoudre ici consiste donc à minimiser le coût total donné par la fonction F (1) sujet aux contraintes (2)-(5).

ALGORITHME DE RESOLUTION

Le problème énoncé au chapitre précédent peut se résumer à la forme:

$$\text{Min } F(v) \qquad (P)$$

$$\text{sà: } v \in \Omega$$

Les contraintes (2)-(5) étant linéaires, Ω est un polytope qu'on va supposer non-vide. En prenant des fonctions de coût sur les arcs strictement positives, on peut supposer Ω compact (en se restreignant aux chemins sans circuits).

Nous allons supposer F convexe de classe C^1 sur un ouvert contenant Ω. Ces hypothèses sont par exemple satisfaites pour des fonctions de coût unitaires du type:

$$s_a^p(v) = \sum_i \alpha_i (\beta_i + v_a^p)^{z(i)} \quad , \ \alpha_i > 0, \ \beta_i > 0, \ 0 \leq z(i) < \infty$$

(idem sur les transferts). Comme le montre cet exemple, les coûts unitaires n'ont pas nécessairement besoin d'être convexes pour que F le soit.

La taille des problèmes rencontrés dans le contexte des réseaux de transport empêche l'utilisation de méthodes du second ordre (utilisant le Hessien ou une approximation) pour résoudre (P). La méthode du premier ordre la plus populaire pour ce genre de problème est celle de Frank-Wolfe (FW). Outre son taux de convergence linéaire peu désirable, cette méthode présente, dans le contexte du problème qui nous intéresse, une difficulté d'ordre technique: la direction réalisable de descente est un vecteur de la taille de v, c'est-à-dire de dimension $n_P(n_A+n_T)$, ce qui peut vite devenir prohibitif en ce qui concerne l'espace mémoire.

Il faut donc tenter de réduire la taille du vecteur de direction de descente. Ω se décompose naturellement en un produit direct $\overline{\prod}\ \Omega^p$, Ω^p représentant l'ensemble des flots du produit p sur les sous-réseaux $m(p)\subset M(p)$. La direction de descente de FW est trouvée via la minimisation d'une approximation linéaire de F. Ce dernier problème se ramène à la résolution de n_P sous-problèmes. On constate donc que le problème (P) possède une structure sous-jacente respectant les produits.

L'algorithme que nous allons présenter, qui nous a été suggéré par Michael Florian et Heinz Spiess, tient compte de ces observations en ce sens que, pour chaque produit $p\in P$, une direction de descente est trouvée, cette direction ayant pour dimension (n_A+n_T). Les variables qui ne sont pas reliées au produit courant sont tenues fixes lors du processus.

Cette approche s'apparente à la méthode de Gauss-Seidel par blocs de variables, à l'exception qu'ici les sous-problèmes ne sont pas résolus exactement. En effet, pour chaque produit p, on n'effectue qu'une seule itération de l'algorithme de FW.

ALGORITHME A

0. INITIALISATION

v (solution initiale)

1. CYCLE MAJEUR

$v_{old} = v$

Pour chaque $p \in P$, faire 1p.

1p. CYCLE MINEUR

$c^p(v)$ (coûts marginaux)

w^p (point extrême (LP))

$d^p := w^p - v^p$ (direction de descente)

λ_p (pas)

$v^p := v^p + \lambda_p d^p$ (mise-à-jour)

2. ARRET

Si $(v \neq v_{old})$ aller à 1.

Commentons un peu plus en détails chacune des étapes principales de cette procédure.

Le cycle mineur 1p consiste à résoudre partiellement le problème (P) restreint au sous-espace des flots du produit p. Une direction de descente est trouvée en minimisant l'approximation linéaire de $F\big|_p$ sur Ω^p, c'est-à-dire:

$$\text{Min } \nabla F^p(v)y \qquad \text{(LPp)}$$

$$\text{sà. } y \in \Omega^p$$

Rappelons que $F(v) = s^T(v)v$. On en conclut que $\nabla F(v) = s^T(v) + v^T s'(v)$. Notons $c(v) \doteq s(v) + s'^T(v)v$ la fonction de *coût marginal*. Le sous-problème linéaire (LPp) s'écrit alors:

$$\text{Min } c^{pT}(v)y$$
$$\text{sà. } y \in \Omega^p$$

Une solution extrémale w^p est obtenue en affectant les demandes $g^{m(p)}$ sur les plus courts chemins des sous-réseaux correspondants aux modes de $m(p)$, plus courts au sens des coûts marginaux $c^p(v)$.

Le pas λ_p est obtenu de manière habituelle, c'est-à-dire par une minimisation unidimensionnelle de F dans la direction $d = (0,..,d^p,..,0)^T$. Si $c^{pT}d^p = 0$ (v^p est une solution de (LPp)), alors on posera $\lambda_p = 0$.

Une solution initiale (extrémale) est obtenue en effectuant un cycle majeur à partir des coûts marginaux à flot nul $c(0)$ et en posant le pas λ_p à un dans chaque cycle mineur 1p.

L'arrêt de la procédure intervient lorsqu'aucune modification n'a été apportée au point courant lors d'un cycle majeur. La solution courante est alors optimale, comme nous allons le voir au chapitre suivant. Pratiquement, l'arrêt de la procédure peut evidemment être basé sur le nombre de cycles majeurs ou sur le temps CPU maximal autorisé. On peut, en outre, utiliser la convexité de F pour construire un critère basé sur le "gap relatif":

$$100 * \frac{F(v) - BLB(v)}{F(v)}$$

où BLB(v) représente la meilleure borne inférieure obtenue au cours des cycles majeurs précédents. Cette borne inférieure se calcule en résolvant tout d'abord le problème linéaire suivant:

$$\text{Min} \quad \nabla F(v)y$$

$$\text{sà.} \quad y \in \Omega$$

Une borne inférieure à la valeur optimale $F(v^*)$ est alors donnée par $F(v) + \nabla F(v)(y^* - v)$, où y^* est une solution optimale du problème linéaire ci-dessus. Il est à noter que cette borne ne peut être calculée par parties lors de chaque cycle mineur 1p. Sa détermination nécessite donc autant de recherches de plus courts chemins qu'il n'y a de matrices de demande à affecter. L'utilisation du gap comme critère d'arrêt doit donc être utilisée avec prudence.

ANALYSE DE CONVERGENCE

Nous allons dans ce chapitre démontrer la convergence globale de l'algorithme A. Pour ce faire, nous allons montrer que F représente une fonction de descente relative à A et que l'application algorithmique, qu'on dénotera A elle aussi, est fermée sur Ω.

Lemme 1: Soit y un point obtenu de v après un cycle mineur $\overline{1p}$. Alors $F(y) \leq F(v)$.

Démonstration:

$$y^p = \begin{cases} v^p + \lambda_p d^p & \text{si } p=\bar{p} \\ v^p & \text{sinon} \end{cases}$$

Donc $y = v + \lambda_p D$, où: $D = (D^p)_{p\in P}$ et $D^p = \begin{cases} d^p = w^p - v^p & \text{si } p=\bar{p} \\ 0 & \text{sinon} \end{cases}$.

Soit $g(\lambda) = F(v+\lambda D)$. g est continuement différentiable.

$$g'(\lambda) = \nabla F(v+\lambda D)D = c^T(v+\lambda D)D$$

$$= \sum_{p\in P} c^{pT}(v+\lambda D)D^p = c^{pT}(v+\lambda D)d^p$$

$$g'(0) = c^{pT}(v)d^p$$

Si $g'(0)=0$, alors on a choisi $\lambda_p=0$ et ainsi $y=v$; donc $F(y)=F(v)$.

Si $g'(0)\neq 0$, alors $g'(0)<0$ par définition de w^p. Donc D est une direction de descente réalisable pour F en v. Par définition de λ_p, on a alors $F(y)<F(v)$. ■

Soit $S\subseteq\Omega$ l'ensemble des points de Kuhn-Tucker du problème (P), c'est-à-dire ici l'ensemble des optima globaux. Si F est strictement convexe, alors S est un singleton. S peut être caractérisé sous la forme variationnelle:

$$S = \{ v\in\Omega : \nabla F(v)(w - v) \geq 0 \quad , \forall w\in\Omega \}$$

Soit $A: \Omega \longrightarrow \Omega$ l'application algorithmique (multivoque).

Lemme 2: $v \in S$ ssi $v \in A(v)$

Démonstration:

Montrons tout d'abord la nécessité de la condition.

$$\nabla F(v)(w - v) \geq 0 \quad , \forall w \in \Omega$$

$$\sum_{p \in P} c^{pT}(v)(w^p - v^p) \geq 0 \quad , \forall w \in \Omega \qquad (*)$$

Fixons $\bar{p} \in P$ et considérons les points $y=(y^p)_{p \in P} \in \Omega$ tels que $y^p=v^p$ si $p \neq \bar{p}$. De l'équation (*), on tire:

$$c^{\bar{p}T}(v)(y^{\bar{p}} - v^{\bar{p}}) \geq 0 \quad , \forall y^{\bar{p}} \in \Omega^{\bar{p}}$$

En particulier, si $w^{\bar{p}}$ est le point extrême généré dans le sous-problème linéaire du cycle mineur $1\bar{p}$, on a que $c^{\bar{p}T}(v)(w^{\bar{p}} - v^{\bar{p}}) = 0$. Donc v ne sera modifié dans aucun cycle mineur et ainsi $v \in A(v)$.

Pour démontrer la suffisance de la condition, nous allons procéder par contraposition. Soit $v \in \Omega - S$. Il existe alors un $w \in \Omega$ tel que:

$$\nabla F(v)(w - v) < 0$$

ou: $$\sum_{p \in P} c^{pT}(v)(w^p - v^p) < 0$$

Il existe donc un produit $\bar{p}$ pour lequel on ait:

$$c^{\bar{p}T}(v)(w^{\bar{p}} - v^{\bar{p}}) < 0$$

En particulier, il est impossible que $c^{pT}(v)d^p=0$ pour tout $p \in P$. Or, dans la preuve du lemme 1, on a remarqué que si $c^{pT}d^p<0$ lors d'un cycle mineur, alors on observe une décroissance stricte de la fonction F, et comme F ne peut augmenter lors d'un cycle mineur, on a que $F(y)<F(v)$ pour tout $y \in A(v)$. En particulier, v n'appartient pas à A(v). ■

Corollaire 1: F est une fonction de descente relativement à A.

Démonstration:

Soit $y \in A(v)$. En applicant n_P fois le lemme 1, on obtient que $F(y) \leq F(v)$. De plus, on a vu dans la preuve du lemme 2 que $F(y) < F(v)$ si $v \in \Omega - S$. ■

La deuxième étape de la preuve consiste à montrer que l'application algorithmique A est fermée. Pour ce faire, supposons que $P=(p_1, \ldots, p_{n_P})$ et décomposons A ainsi:

$$A = A^{p_{n_P}} \circ \ldots \circ A^{p_1} \quad : \quad \Omega \longrightarrow \Omega$$

$$\text{où:} \quad A^p = M \circ D^p \quad : \quad \Omega \longrightarrow \Omega$$

A^p est l'application algorithmique correspondant à un cycle mineur, D^p représente la recherche d'une direction de descente et M la minimisation unidimensionnelle. Plus formellement, posons:

$$D^p : \Omega \longrightarrow \Omega \times U$$

$$v \longrightarrow D^p(v)$$

où: $D^p(v) = \{(v,d): d^T=(0,..,d^{pT},..,0),\ d^p=w^p-v^p \text{ où: } w^p \in \underset{y \in \Omega^p}{\text{Arg Min}}\ \nabla F(v)^p y\}$

et: $U = \Omega - \Omega = \{y: y = y1 - y2,\ y1,y2 \in \Omega\}$

$$M : \Omega \times U \longrightarrow \Omega$$

$$(v,d) \longrightarrow M(v,d)$$

où: $M(v,d) = \{y: y = \begin{cases} v \text{ si } \nabla F(v)d=0 \\ v+\lambda^* d \text{ sinon, où } \lambda^* \in \underset{0 \leq \lambda \leq 1}{\text{Arg Min}}\ F(v+\lambda d) \end{cases}\}$

Lemme 3: D^p est fermée sur Ω.

Démonstration:

Soit $\{v^k\}_k$ une suite de Ω qui converge vers v. Soit d'autre part $\{(v^k,d^k)\}_k$ une suite de $\Omega x U$ telle que $(v^k,d^k) \varepsilon D^p(v^k)$ et qui converge vers (v,d). De la définition de $D^p(v^k)$, on peut écrire d^k comme $y^k - v^k$ où $y^k = v^k$ sauf pour la p-ième composante où on a $(y^k)^p = (w^k)^p$, $(w^k)^p$ étant une solution du sous-problème linéaire. On a donc:

$$y^k = d^k + v^k \quad , \quad y^k \varepsilon \Omega$$

Le terme de droite tend vers $d+v=y$; $y=v$ sauf pour la p-ième composante. Pour montrer que (v,d) appartient à $D^p(v)$, il suffit alors de montrer que y^p est une solution optimale du sous-problème linéaire:

$$\text{Min } \nabla F(v)^p z$$
$$\text{sà. } z \varepsilon \Omega^p$$

Or, on a: $\nabla F(v^k)^p (y^k)^p \leq \nabla F(v^k)^p z^p \quad , \forall z^p \varepsilon \Omega^p$

En utilisant la continuité de ∇F, on en conclut que:

$$\nabla F(v)^p y^p \leq \nabla F(v)^p z^p \quad , \forall z^p \varepsilon \Omega^p \quad \blacksquare$$

Lemme 4: M est fermée sur $D^p(v)$, $\forall v \varepsilon \Omega$.

Démonstration:

Soit $(v,d) \varepsilon D^p(v)$ et soit $\{(v^k,d^k)\}_k$ une suite de $\Omega x U$ qui converge vers (v,d). Soit d'autre part $\{y^k\}_k$ une suite de Ω telle que $y^k \varepsilon M(v^k,d^k)$ et qui converge vers y. $y \varepsilon \Omega$ car Ω est compact. Il faut montrer que $y \varepsilon M(v,d)$.

Examinons tout d'abord le cas $d \neq 0$. On peut alors supposer qu'il existe K tel que $d^k \neq 0$ pour tout $k \geq K$. Dans la suite, nous supposerons que $k \geq K$.

$$y^k = v^k + \lambda^{*k} d^k$$

$$\text{Donc: } \lambda^{*k} = \frac{||y^k - v^k||}{||d^k||}$$

Le terme de droite tend vers:

$$\frac{||y - v||}{||d||} = \lambda^* \in [0,1]$$

ainsi: $y = v + \lambda^* d$

Il reste à voir que le λ^* ainsi trouvé résoud le problème de minimisation unidimensionnelle. On a que $F(y^k) \leq F(v^k + \lambda d^k)$ pour tout $\lambda \in [0,1]$. En utilisant la continuité de F, en ayant fixé λ on passe à la limite et on obtient:

$$F(y) = F(v + \lambda^* d) \leq F(v + \lambda d) \quad , \quad \forall \lambda \in [0,1]$$

Il reste maintenant à examiner le cas $d=0$.

$$y^k - v^k = \lambda^{*k} d^k$$

Comme $\lambda^{*k} \in [0,1]$ pour tout k et comme d^k tend vers 0, le terme de droite tend lui aussi vers 0. En passant à la limite, on obtient donc que $y=v$. ■

Corollaire 2: A^P est fermée sur Ω.

Démonstration:

$$A^P : \Omega \longrightarrow \Omega \times U \longrightarrow \Omega$$

$\Omega \times U$ est clairement compact puisque Ω l'est. Grâce aux lemmes 3 et 4, toutes les hypothèses du lemme 4.2 de Zangwill [1] sont alors satisfaites. ■

Corollaire 3: A est fermée sur Ω.

Démonstration:

Corollaire 2 et lemme 4.2 de Zangwill [1]. ■

Du théorème A de convergence de Zangwill [1], on déduit alors des corollaires 1 et 3 la convergence globale de la procédure A.

RESULTATS

Nous ne présenterons dans ce chapitre que les résultats de tests préliminaires obtenus lors du développement du logiciel implantant la procédure A. En particulier, il faut considérer avec précautions les résultats concernant la performance en temps CPU (M68000). Nous allons présenter des comparaisons entre notre procédure et l'algorithme de Frank-Wolfe au niveau de la décroissance de la fonction objectif versus le nombre d'itérations (cycles majeurs) et le temps CPU.

Les résultats qui suivent proviennent de variantes du problème (P) sur

un même réseau de base formé de 5 origines, 5 destinations, 19 noeuds réguliers, 116 liens et 226 transferts. Sept modes se partagent ce réseau.

Dans le premier problème, un seul produit est transporté dans le réseau par tous les modes. Une seule matrice décrit la demande. Il est clair que , dans ce cas, les deux procédures sont équivalentes, ce qui est confirmé dans la figure 6. La légère différence au niveau du temps de calcul provient du fait que pour l'algorithme A, le gap est recalculé à la fin d'un cycle majeur, alors que pour le cas de FW, ce gap est automatiqument disponible au cours de l'itération.

Dans le second problème, trois produits sont affectés sur le réseau, deux d'entre eux pouvant emprunter tous les modes, le troisième étant restreint à un sous-ensemble de trois modes. Les demandes pour chacun des produits sont décrites par une seule matrice. Contrairement au premier problème, les deux procédures sont ici distinctes, l'algorithme A présentant un comportement légèrement plus efficace que FW.

Le troisième problème contient cinq produits dont deux seulement peuvent être transportés par tous les modes. La demande pour le premier produit est décomposée en trois matrices, chacune étant affectée sur différents sous-réseaux. De même, la demande du deuxième produit est répartie sur deux matrices. Comme le démontre la figure 8, les performances relatives de notre procédure sont très bonnes.

Finalement, le quatrième problème consiste à transporter les mêmes cinq produits que ceux du problème précédent avec les mêmes matrices de demandes. La différence est qu'ici, tous les modes sont permis. De plus,

la fonction de coût généralisé est composée d'une somme de trois coûts (opération, délai, consommation d'énergie) alors que pour les trois premiers problèmes, seul le délai était considéré. Ceci a pour conséquence de tripler le temps d'évaluation des fonctions de coût unitaire, ce qui, pour un problème de la taille de celui utilisé, représente la quasi-totalité du temps CPU de l'algorithme.

Dans tous les cas où le nombre de produits est supérieur à un, la procédure A présente une amélioration plus ou moins marquée relativement à l'algorithme de FW aussi bien au niveau du nombre d'itérations que du temps de calcul. Grâce à l'économie d'espace mémoire réalisée dans notre procédure, cette dernière pourra de plus résoudre des problèmes de plus grande taille.

CONCLUSION

La procédure d'affectation multimode-multiproduit présentée ici consiste en une modification de l'algorithme de Frank-Wolfe qui permet le traitement de problèmes de plus grande taille et qui semble présenter de meilleures performances tant au niveau du nombre d'itérations qu'au niveau du temps de calcul.

REFERENCES

[1] Zangwill W.I. Nonlinear Programming. A Unified Approach. Prentice Hall, Englewood Cliffs, N.J. (1969).

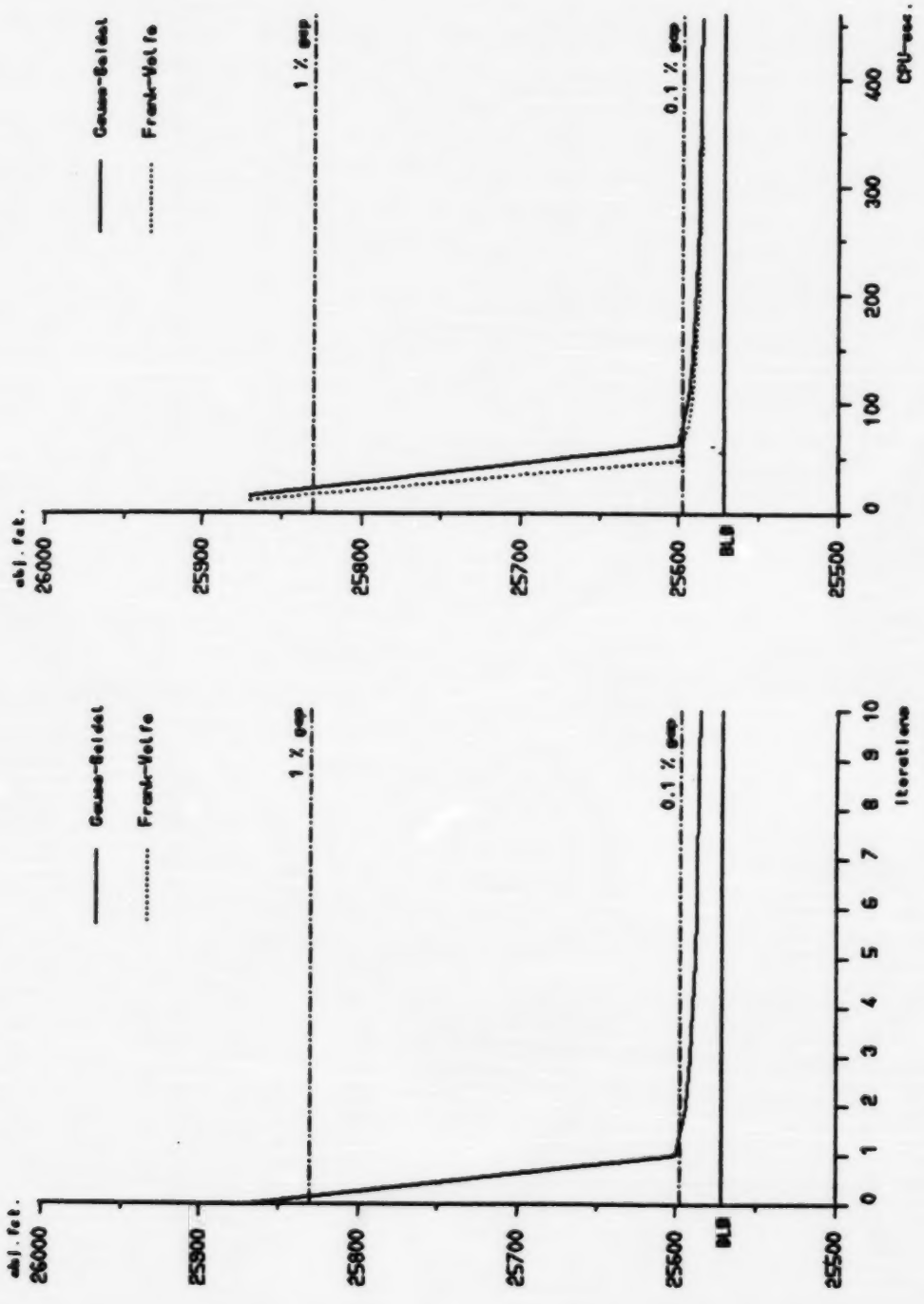

Figure 6. Problème 1.

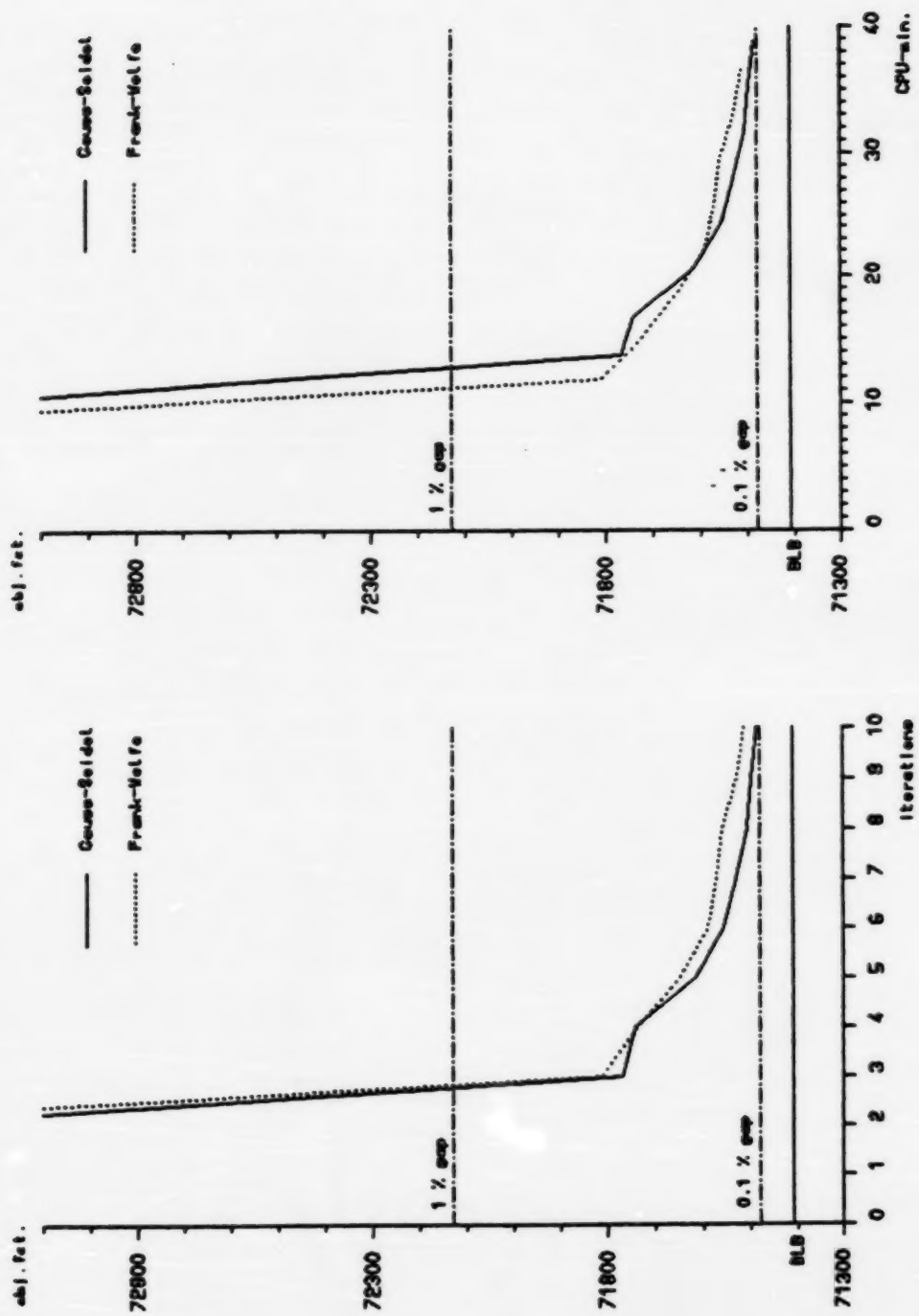

Figure 7. Problème 2.

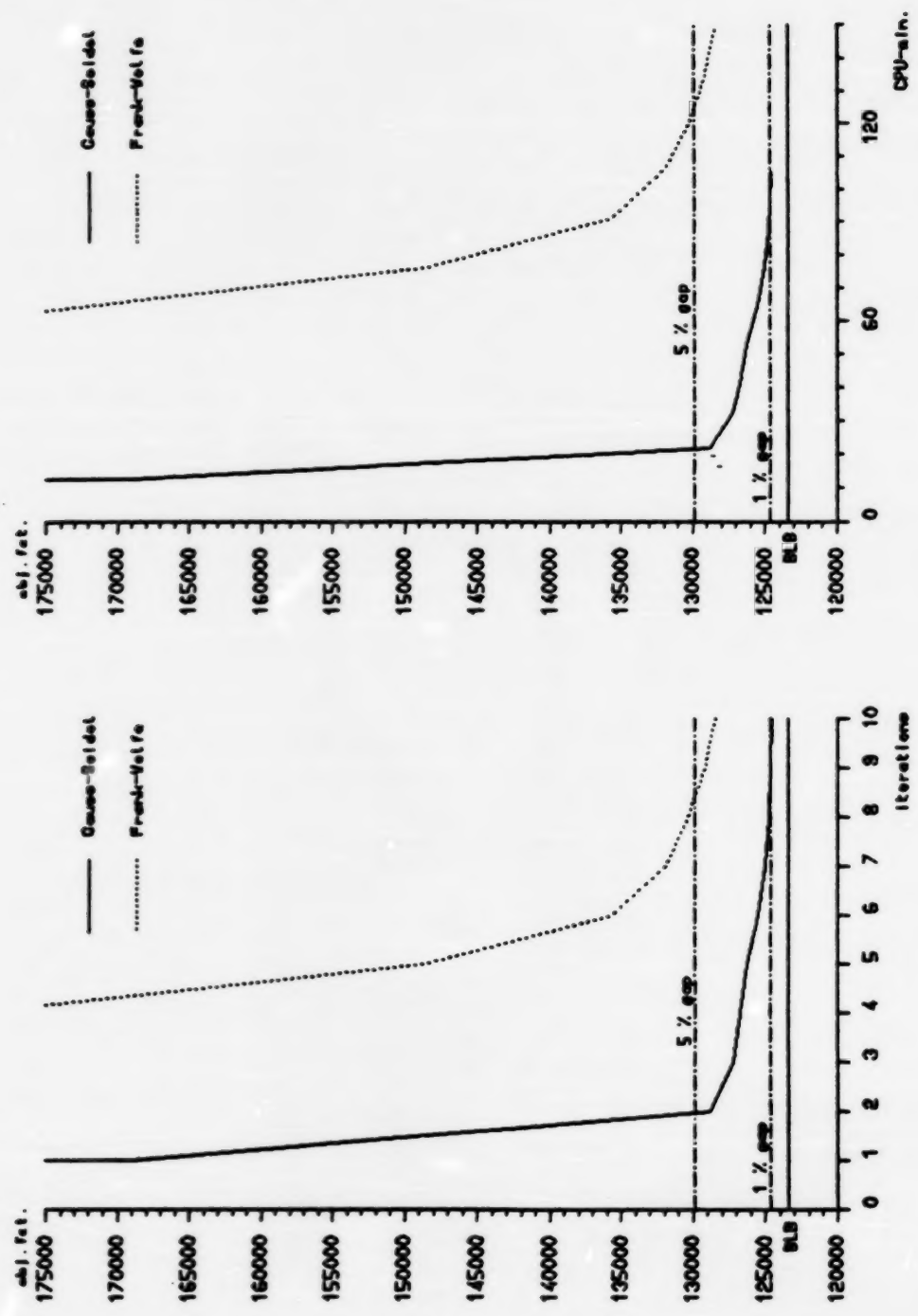

Figure 8. Problème 3.

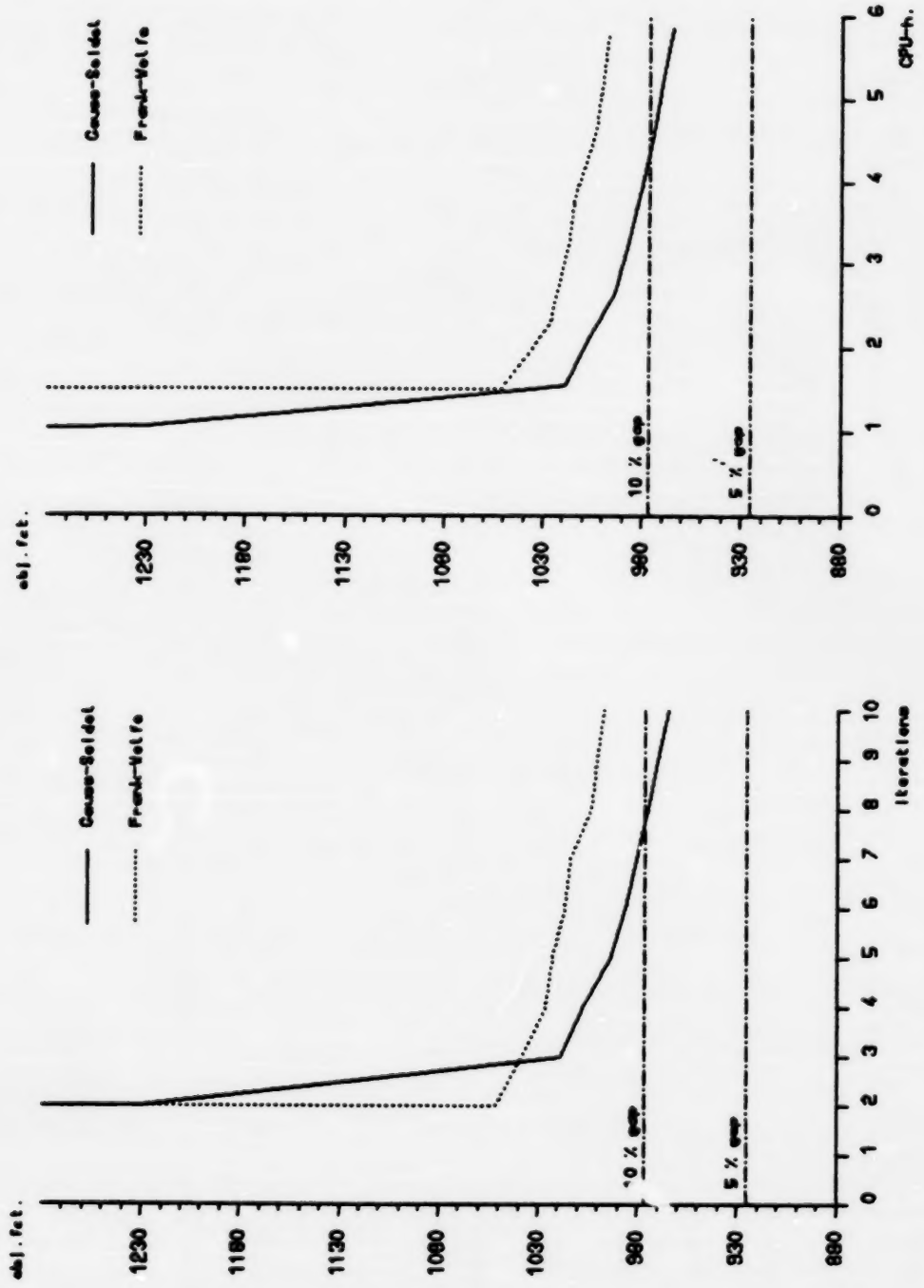

Figure 9. Problème 4.